SUPPLEMENTAL MATERIAL FOR “TOWARDS MAKING UNSUPERVISED GRAPH HASHING ROBUST”

Anonymous ICME submission

In this supplementary material, we introduce ablation studies in support of the primary text, analyze convergence behavior, and provide a detailed proof for Theorem 1. In the ablation studies, we respectively study the impact of dual-graph and the impact of DGRH’s parameters γ_1 and γ_2 over the retrieval performance.

0.1. Ablation studies

Effects of graph. To construct Laplacian graphs, we firstly adopt the k-means method to generate the anchor point set and set $\{m_s = 2000, s = 2, m_f = 300\}$ for MNIST and Caltech-256 datasets where s is the number of nearest neighbors used in k-means. Meanwhile, we set $\{m_s = 300, s = 3, m_f = 50\}$ for CIFAR-10 dataset. In this paper, DGRH preserves the similarity of images with two types graph. To validate the effectiveness of the image similarity preservation for hashing performance, we compare our method with a variant of our method DGRH-G which removes the dual-graph part. We conduct experiments on three datasets with the hash code length varied from 16 bits to 128 bits respectively. The retrieval results about mAP are shown in Table 1 and Fig. 1. From Fig. 1, we find that on all three datasets, our method DGRH outperforms DGRH-G. The superior performance is mainly because two types of graphs make the learned low-rank matrix preserve more structure information.

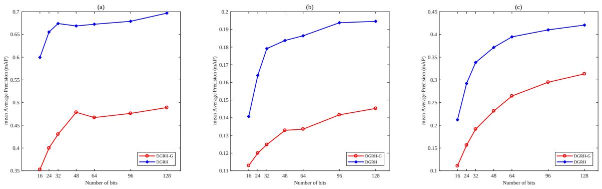

Fig. 1. The mAP of DGRH and DGRH-G versus different numbers of bits (a) MNIST, (b) CIFAR-10, and (c) Caltech-256, respectively

Parameter Sensitivity. To further observe the performance variations of graph with parameters γ_1 and γ_2, we conduct empirical experiments on three datasets with the code length fixed to 64 bits. We vary the value of them from the range of $\{\gamma_1, \gamma 2\} \in \{5^i\}_{i=-4}^{2}$, and experimental results are shown in Fig. 2. From Fig. 2, we can find that the performance is relatively high when γ_1 is in the range of $\{0.5,\ 5,\ 50\}$ and γ_2 is in the range of $\{0.0005, 0.005, 0.05, 0.5\}$ for MNIST, CIFAR-10 and Caltech-256. We also find that DGRH can obtain good results on all three datasets when γ_1 is about 0.5.

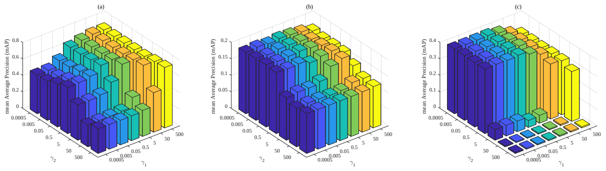

Fig. 2. Parameter sensitivity on (a) MNIST, (b) CIFAR-10 and (c) Caltech-256 datasets

0.2. Convergency analysis.

To analyze the convergency of DGRH, we conduct experimental analysis on MNIST, CIFAR-10 and Caltech-256 with the code length 64 bits fixed. It should be noted that similar results can be obtained on other hash code lengths. The convergency curves are shown in Fig. 6. It can be easily observed from the figure that DGRH can converge very fast within only a few iterations and the objective function value does not change significantly after several iterations on three datasets. Those results empirically validate that the convergence of DGRH can be achieved with the proposed optimization method.

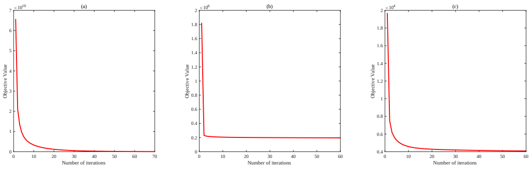

Fig. 3. Effets of different numbers of training data in DGRH model on (a) MNIST, (b) CIFAR-10, and (c) Caltech-256, respectively

0.3. Proof for Theorem 1

In this section, we analyze the reason why DGRH is stable for learning the hash code B according to [1]. Let $\varPhi = (Z, Y) =$

Table 1. The mAP of DGRH and DGRH-G on MNIST, CIFAR-10,and Caltech-256, respectively.

Method	Datasets	Bits						
		16	24	32	48	64	96	128
DGRH-G	MNIST	0.3525	0.3999	0.4303	0.4786	0.4671	0.4762	0.4891
	CIFAR-10	0.1129	0.1200	0.1248	0.1329	0.1336	0.1416	0.1453
	Caltech-256	0.1107	0.1562	0.1915	0.2314	0.2645	0.2948	0.3133
DGRH	MNIST	0.5994	0.6554	0.6740	0.6687	0.6725	0.6790	0.6969
	CIFAR-10	0.1407	0.1640	0.1792	0.1837	0.1864	0.1938	0.1946
	Caltech-256	0.2123	0.2921	0.3384	0.3715	0.3947	0.4101	0.4207

$\{z_i, y_i\}_{i=1}^n$ be the training samples for DGRH and $\varPhi^*$ be the samples with the ith example $\phi_i = (z_i, y_i), i = 1, ..., n$ in $\varPhi$ replaced with an i.i.d. one $\phi_i' = (z_i', y_i')$. We regard an algorithm as a stable one if the output hashing codes do not change much when a training sample is deleted or replaced with an independent and identically distributed distribution.

Definition 1. A hashing coding algorithm is $\beta(n)$-stable if the following holds

$$\forall \varPhi, \varPhi^*, \phi_i, \phi_i', i = 1, ..., n, \quad \|B(\varPhi) - B(\varPhi^*)\|_F \le \beta(n)$$

where $B(\varPhi)$ and $B(\varPhi^*)$ are the hash codes learned by employing $\varPhi$ and $\varPhi^*$, respectively, and $\beta(n)$ converges to zero with respect to the sample size n.

In this paper, the proposed method DGRH is optimized by employing an alternating iteration strategy, we assume that the optimization algorithm stops with K iterations and in the kth iteration, where $k = 1, ..., K$, and B^k, W^k, X^k, $E^k$are obtained. We prove that DGRH for learning W is stable in each iteration because of the l_2-regularizaion $\|W\|_F^2$.

We first modify the objective function Eq. (3) to ensure that the regularization parameters are invariant to the sample size n, class size c, and code length k. The modified model is as follows:

$$\frac{\lambda_2^*}{nk}\left\|B - W^TX\right\|_F^2 + \frac{\lambda_3^*}{ck}\|W\|_2^2 + \frac{1}{nk}\varGamma(X) \qquad (1)$$

where $\varGamma(X) = \frac{1}{2}\|Z - X - E\|_F^2 + \lambda_1\|E\|_1 + \gamma_1 tr(XL_1X^T) + \gamma_2 tr(X^TL_2X)$, $\lambda_2 = 2\lambda_2^*$, $\lambda_3 = \frac{2n\lambda_3^*}{c}$. We now prove that DGRH is stable with respect to W in each iteration.

Note that Bousquet and Elisseeff [2] proved that stable algorithms will generalize well and that our empirical results in Section 3 support our theoretical analysis by showing that the newly proposed methods generalize well on the test samples.

0.3.1. Upper Bound of X

Before proofing Theorem 1, we get the upper bound of $\|x_i\|_2$ .Setting E to zero, we can get the following formulation:

$$\frac{1}{2}\|Z - X\|_F^2 + \gamma_1 tr(XL_1X^T) + \gamma_2 tr(X^TL_2X) \le \frac{1}{2}\|Z\|_F^2$$

Let δ be the minimum eigervalue of L_2, we can get $tr(X^TL_2X) \ge \delta tr(X^TX)$. Thus,

$$\frac{1}{2}\|X\|_F^2 + \gamma_1 tr(XL_1X^T) + \gamma_2\delta tr(X^TX) \le tr(Z^TX)$$

then we can get

$$tr(X\varPsi X^T) \le tr(Z^TX), \qquad (2)$$

where $\varPsi = \frac{1}{2}I + \gamma_1 L_1 + \gamma_2\delta I.$

Defining $Z^TX = Z^TX\varPsi^{\frac{1}{2}}\varPsi^{-\frac{1}{2}}$, we can obtain

$$\left\|X\varPsi^{\frac{1}{2}} - \frac{1}{2}\varPsi^{-\frac{1}{2}}Z^T\right\|_F^2 \le \left\|\frac{1}{2}\varPsi^{-\frac{1}{2}}Z^T\right\|_F^2$$

Let ξ be the minimum eigenvalue of $\varPsi$, we can get

$$\|X\|_F \le \frac{1}{\xi}\|Z\|_F \qquad (3)$$

Assuming $\|Z\|_F \le \varTheta$, finally, the upper bound of $\|x_i\|_2$ can be represented by:

$$\|x_i\|_2 \le \frac{\varTheta}{n\xi}, \forall i = 1, ..., n \qquad (4)$$

0.3.2. Bregman Matrix Divergence

As the proof shown in [1], we need the Bregman matrix divergence [3] to prove **Theorem 1**.

Definition 2. For any matrix A and B of the same size, the bregman matrix divergence with respect to function f is defined as:

$$Bgm_f(A, B) = f(A) - f(B) - tr(\nabla f(B)^T(A - B)) \quad (5)$$

where $\nabla f(B)$ denotes the derivative of f as B.

It is proven that if function f is convex, the Bergman divergence will be non-negative and additive. For example, $Bgm_f(A, B) \ge 0$, and $Bgm_{f+g}(A, B) = Bgm_f(A, B) + Bgm_g(A, B)$, if f and g are both convex.

0.3.3. Proof for Theorem 1

According to the non-negative and additive properties of Bregman divergence, we have

$$\begin{aligned} &Bgm_{f_{\mathit{\Phi}}}(Q,P)+Bgm_{f_{\mathit{\Phi}^*}}(P,Q)\geq \\ &\quad Bgm_{r_{\mathit{\Phi}}}(Q,P)+Bgm_{r_{\mathit{\Phi}^*}}(P,Q) \end{aligned} \tag{6}$$

where $Q=W^k(\mathit{\Phi}^*)$, and $P=W^k(\mathit{\Phi})$

$$\begin{aligned} &Bgm_{r_{\mathit{\Phi}}}(Q,P)+Bgm_{r_{\mathit{\Phi}^*}}(P,Q)= \\ &\frac{2\lambda_3^*}{ck}\|P\|_F^2-\frac{2\lambda_3^*}{ck}\|Q\|_F^2-\frac{2\lambda_3^*}{ck}tr(Q^T(P-Q))+ \\ &\frac{2\lambda_3^*}{ck}\|Q\|_F^2-\frac{2\lambda_3^*}{ck}\|P\|_F^2-\frac{2\lambda_3^*}{ck}tr(P^T(Q-P)) \\ &=\frac{2\lambda_3^*}{ck}\|Q-P\|_F^2 \end{aligned} \tag{7}$$

$$\begin{aligned} &Bgm_{f_{\mathit{\Phi}}}(Q,P)+Bgm_{f_{\mathit{\Phi}^*}}(P,Q)= \\ &\frac{\lambda_2^*}{nk}\left\|B-P^TX^i\right\|_F^2+\frac{2\lambda_3^*}{ck}\|P\|_F^2+\frac{1}{nk}F(X^i)- \\ &\frac{\lambda_2^*}{nk}\left\|B-Q^TX^i\right\|_F^2-\frac{2\lambda_3^*}{ck}\|Q\|_F^2-\frac{1}{nk}F(X^i)+ \\ &\frac{\lambda_2^*}{nk}\left\|B-Q^TX^i*\right\|_F^2+\frac{2\lambda_3^*}{ck}\|Q\|_F^2+\frac{1}{nk}F(X^i*)- \\ &\frac{\lambda_2^*}{nk}\left\|B-P^TX^i*\right\|_F^2-\frac{2\lambda_3^*}{ck}\|P\|_F^2-\frac{1}{nk}F(X^i*) \\ &=\frac{\lambda_2^*}{nk}\left\|B-P^TX^i\right\|_F^2-\frac{\lambda_2^*}{nk}\left\|B-P^TX^i*\right\|_F^2+ \\ &\frac{\lambda_2^*}{nk}\left\|B-Q^TX^i*\right\|_F^2-\frac{\lambda_2^*}{nk}\left\|B-Q^TX^i\right\|_F^2 \\ &=\frac{\lambda_2^*}{nk}\left\|b-P^Tx^i\right\|_2^2-\frac{\lambda_2^*}{nk}\left\|b-P^Tx^i*\right\|_2^2+ \\ &\frac{\lambda_2^*}{nk}\left\|b-Q^Tx^i*\right\|_2^2-\frac{\lambda_2^*}{nk}\left\|b-Q^Tx^i\right\|_2^2 \\ &\leq\frac{2M\lambda_2^*}{nk}\left\|(Q^T-P^T)x^i\right\|_2+\frac{2M\lambda_2^*}{nk}\left\|(Q^T-P^T)x^{i*}\right\|_2 \end{aligned} \tag{8}$$

where the first equality holds because $\nabla_W f_{\mathit{\Phi}^*}(Q)=\nabla_W f_{\mathit{\Phi}}(P)=0$.

Combining the results Eq. 4, Eq. 7 and Eq. 8, we have

$$\left\|W^k(\mathit{\Phi}^*)-W^k(\mathit{\Phi})\right\|_F\leq\frac{2Mc\lambda_2^*\mathit{\Theta}}{n^2\xi\lambda_3^*} \tag{9}$$

This completes the proof.

1. REFERENCES

[1] Jie Gui, Tongliang Liu, Zhenan Sun, Dacheng Tao, and Tieniu Tan, "Fast supervised discrete hashing," *IEEE transactions on pattern analysis and machine intelligence*, vol. 40, no. 2, pp. 490–496, 2017.

[2] Olivier Bousquet and André Elisseeff, "Stability and generalization," *Journal of machine learning research*, vol. 2, no. Mar, pp. 499–526, 2002.

[3] G Alistair Watson, "Characterization of the subdifferential of some matrix norms," *Linear algebra and its applications*, vol. 170, pp. 33–45, 1992.